

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Domen Lipovšek

Vizualizacija mednarodne blagovne menjave v Sloveniji

DIPLOMSKO DELO

VISOKOŠOLSKI ŠTUDIJSKI PROGRAM
PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: viš. pred. dr. Alenka Kavčič

Ljubljana, 2017

COPYRIGHT. Rezultati diplomske naloge so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavo in koriščenje rezultatov diplomske naloge je potrebno pisno privoljenje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Vizualizacija mednarodne blagovne menjave v Sloveniji

Tematika naloge:

Med javnimi odprtimi podatki, ki so v zadnjih letih v skladu z direktivo EU prosto dostopni v Sloveniji, so tudi podatki o slovenski blagovni menjavi z drugimi državami sveta. Za lažje razumevanje tako kompleksnih podatkov potrebujemo primeren prikaz podatkov v grafični obliki, ki s pomočjo interakcije omogoča tudi poglobljen vpogled v posamezne podatke. V okviru diplomske naloge zasnujte in izdelajte spletno aplikacijo za vizualizacijo mednarodne blagovne menjave na podlagi odprtih podatkov slovenskega uvoza in izvoza, ki jih nudi Statistični urad Republike Slovenije. Aplikacija naj na razumljiv način predstavi blagovno menjavo Slovenije s posameznimi državami sveta, omogoča pa naj tako globalni pogled kot tudi podrobnosti blagovne menjave na nivoju posameznih držav. Pri zasnovi in realizaciji aplikacije uporabite sodobne spletne tehnologije in orodja.

Rad bi se zahvalil svoji mentorci prof. dr. Alenki Kavči in prof. dr. Matiji Maroltu strokovno pomoč, svetovanje in smernice pri izdelavi in pisanju diplomske naloge.

Kazalo

Povzetek

Abstract

1	Uvod	1
1.1	Vizualizacija podatkov	1
1.2	Odprti podatki	4
1.3	Vizualizacije odprtih podatkov	5
1.4	Vizualizacije odprtih podatkov na slovenskem področju	7
1.5	Vizualizacije podatkov na temo uvoza in izvoza	9
1.6	Cilj	13
2	Pregled tehnologij	15
2.1	Iskalnik Elasticsearch	16
2.2	Format JSON	18
2.3	Knjižica D3.js	19
2.4	Bootstrap	19
3	Vizualizacija mednarodne blagovne menjave v Sloveniji	21
3.1	Podatki	21
3.2	Vizualizacija uvoz in izvoz Slovenije	23
4	Sklepne ugotovitve	31
	Literatura	33

Seznam uporabljenih kratic

kratica	angleško	slovensko
API	application programming interface	programski vmesnik
HTML	HyperText Markup Language	označevalni jezik za oblikovanje večpredstavnostnih dokumentov
JSON	JavaScript Object Notation	objektna notacija JavaScript
CSS	Cascading Style Sheet	kaskadne stilske podloge
DOM	Document Object Model	dokumentno objektni model
SVG	Scalable vector graphics	umerljiva vektorska grafika

Povzetek

Naslov: Vizualizacija mednarodne blagovne menjave v Sloveniji

Avtor: Domen Lipovšek

Cilj diplomskega dela je bil izdelati interaktivno vizualizacijo podatkov z uporabo odprtih podatkov, katere je priskrbel Statistični urad Republike Slovenije. Začetni del govori o področju vizualiziranja podatkov in politiki odprtih podatkov. Predstavimo koristi odprtih podatkov in njihov pomen v modernem svetu. Dotaknemo se tudi vizualizacij odprtih podatkov širše in v slovenskem prostoru ter predstavimo cilj in namen diplomskega dela. Naslednja poglavja predstavijo tehnologije, katere so bile uporabljene pri izdelavi aplikacije. Opišemo zahteve aplikacije in pojasnimo, zakaj smo se odločili za izbrane rešitve. V tem poglavju govorimo o podatkovni bazi Elasticsearch, ogrodju Bootstrap, JavaScript knjižici D3.js in standardu JSON. V končnem poglavju predstavimo izdelano aplikacijo Vizualizacija mednarodne blagovne menjave v Sloveniji. Začnemo z opisom uporabljenih podatkov in nadaljujemo z razčlenitvijo in opisom pogledov vizualizacije. Na koncu se dotaknemo težav, na katere smo naleteli tekom razvoja aplikacije in njihovih rešitev.

Ključne besede: vizualizacija podatkov, Elasticsearch, HTML, CSS, JavaScript, mednarodna blagovna menjava Slovenije, odprti podatki.

Abstract

Title: Visualisation of international trade in goods in Slovenia

Author: Domen Lipovšek

The aim of this thesis was to produce an interactive data visualization using open data provided by the Statistical Office of the Republic of Slovenia. The first part of the thesis presents the field of data visualization and the policy of open data. Further on it explains the benefits of open data and their role in the modern world. The thesis mentions open data visualization abroad and on Slovenian grounds. It elaborates on the aim and purpose of this work. Following chapters focus on the technologies that were used to make an application. Moreover, application demands are described and chosen solutions are explained. The chapter discusses the Elasticsearch database, Bootstrap toolkit, D3.js JavaScript library, and JSON standard. The final chapter introduces the developed application Visualization of the international trading system in Slovenia. It starts with the description of used data and continues with the analysis and description of visualization views. The end deals with problems that occurred during application development and their solutions.

Keywords: data visualization, ElasticSearch, HTML, CSS, JavaScript, international trade in goods in Slovenia, open data.

Poglavje 1

Uvod

Živimo v dobi informacij, kjer dnevno nastajajo ogromne količine novih podatkov. Poleg tega se čedalje bolj uveljavlja trend odprtih podatkov, kar pomeni, da so podatki dostopni vsakomur in jih lahko uporabljamo brez omejitev. Vendar dostop do podatkov ni dovolj, pomembna je tudi njihova pravilna interpretacija oziroma predstavitev, saj so brez le-te neuporabni. Tu naletimo na problem, kako predstaviti podatke na čim bolj enostaven, jasen, uporaben in zanimiv način.

V sodelovanju s Statističnim uradom Republike Slovenije smo razvili spletno interaktivno vizualizacijo podatkov na temo uvoz in izvoz Republike Slovenije. Razvoj se je začel s prenosom in obdelavo podatkov, ki jih je priskrbel Statistični urad. Nato je sledila izgradnja uporabniškega vmesnika, na koncu pa še testiranje aplikacije in reševanje hroščev in napak.

1.1 Vizualizacija podatkov

Podatke, ki jih dnevno ustvarimo, moramo znati pravilno interpretirati oziroma nam morajo biti pravilno predstavljeni. Podatke lahko predstavimo ustno ali grafično, vendar, kot pravi rek: »Slika pove več kot tisoč besed«, je podatke smiselno predstaviti vizualno. Razlog za takšen prijem je v tem, da človek precej lažje in hitreje procesira vizualne podatke od ostalih. Vizu-

alizacija podatkov se ukvarja s predstavljanjem podatkov na grafičen način., vendar moramo ločiti razliko med vizualizacijo in infografiko.

Pod ime infografika spadajo vizualne predstavitve podatkov, ki so ročno izrisane, specifične za podatke (težje jih ponovno ustvarimo z drugimi podatki), estetsko bogate in podatkovno revne [12]. Kontrastno so vizualizacije podatkov izrisane algoritmično, brez težav jih izrišemo ponovno z uporabo drugih podatkov, so estetično manj bogate in zajemajo velike količine podatkov [12]. V delu se osredotočamo na slednje.

Vizualiziranje podatkov sega daleč v zgodovino. Začetki vizualizacije so se začeli z geometrični diagrami, zvezdnimi kartami in zemljevidi za navigacijo. Tu lahko omenimo prve zemljevide, prve grafikone premikanja nebesnih teles skozi čas, prve atlase in podobno. Ob napredku statistike je napredovala tudi vizualizacija podatkov. Za očeta modernih grafikonov štejemo Williama Playfairja, ki je prvi razvil črtni, stolpični in tortni grafikoni. V devetnajstem stoletju je vizualizacija podatkov doživela svoj največji razcvet. V znanstvenih revijah so se začele pojavljati grafične analize naravnih in fizičnih pojavov, pojavili so se prvi geološki zemljevidi, v tem obdobju so bili razviti tudi vsi moderni grafikoni. Z razvojem računalništva in računalniške grafike je vizualizacija podatkov doživela ponoven razcvet. Pojavile so se namenske aplikacije, s svetovnim spletom se je povečal dostop do informacij in podatkov, vizualizacije so dobile element interaktivnosti [3].

Splošno vizualizacije podatkov ločujemo na vizualizacije, ki nam podatke predstavljajo, in vizualizacije, kjer podatke raziskujemo. Vizualizacije za raziskovanje uporabimo, kadar imamo podatke, vendar nismo prepričani, kaj se nahaja v njih. Takšen tip vizualizacije se po navadi uporablja v fazi analize podatkov za iskanje zgodbe, ki jo govorijo podatki. Ko že poznamo zgodbo podatkov, uporabimo vizualizacijo za predstavitev podatkov, da povemo zgodbo naprej. Če je vizualizacija raziskovanja podatkov del faze analize, je vizualizacija predstavljanja del faze predstavljanja podatkov [12]. Vizualizacije podatkov lahko ločimo tudi glede na število podatkovnih dimenzij, ki jih prikazujejo. Večje je število dimenzij, bolj je vizualizacija kompleksna

in zahtevna, tako za izdelavo kot za branje podatkov. Večina vizualizacij nima več kot tri ali štiri dimenzije, čeprav se najdejo tudi izjeme z več kot sedem, osem ali devet. Razlog za omejeno število dimenzij je v težavnosti pri izbiri vizualnih spremenljivk, saj za vsako dimenzijo rabimo svojo vizualno spremenljivko. Z dodajanjem dimenzij povečujemo kompleksnost, saj moramo prikazati tudi povezave med dimenzijami in obenem paziti na videz in funkcionalnost vizualizacije [12].

Ko razmišljamo, kako bi predstavili dimenzijo podatkov, govorimo o vizualnih spremenljivkah. Odgovor na to vprašanje je podal francoski kartograf Jacques Bertin, ki je v svojem delu predstavil *Bertinove vizualne spremenljivke*. Spremenljivke je razdelil na *oznake* in *kanale*. Oznake sestavljajo točke, črte, področje, površine in volumen, kanale pa pozicija, velikost, oblika, svetlost, orientacija, barva, tekstura in premik [1].

Pri sami izdelavi vizualizacije se držimo načela: »prvo funkcionalnost, potem lepota«. Principe izdelave dobrih vizualizacij je opisal profesor Edward Tufte:

- pokaži podatke,
- spodbudi gledalca k razmišljanju o podatkih in ne o metodologiji, grafičnem izgledu, uporabljeni tehnologiji in ostalem,
- ne popači podatkov,
- prikaži čim več števil na majhnem področju,
- veliki podatkovni nabori naj bodo logično povezani,
- spodbujaj primerjavo različnih delov podatkov,
- prikaži podatke na več nivojih, od najširšega do najmanjšega,
- vsaka stvar naj služi namenu (opis, dekoracija, tabelacija),
- poveži s statističnim in besednim opisom podatkov [26].

1.2 Odprti podatki

Pojem odprti podatki definira podatke, ki so dostopni vsakomur in jih lahko uporabljamo brez omejitev. Državni organi posredujejo veliko enostavnih podatkov, ki so lahko družbi, ekonomsko ali socialno, v veliko korist. Zato se čedalje več evropskih držav odloča za odpiranje podatkov za širšo javnost. Evropska unija je leta 2003 sprejela direktivo 2003/98/ES za spodbujanje ponovne uporabe odprtih podatkov v javnem sektorju. Njen cilj je bil zagotoviti enakopraven odnos do vseh potencialnih uporabnikov odprtih podatkov in tako povečati ekonomske koristi in povečati transparentnost [7].

Koristi odprtih podatkov delimo na direktne in indirektne koristi. Direktne koristi so denarne koristi v trgovinskih dejavnostih v obliki prihodkov, števila delovnih mest, vpletenih v storitvi in prihranku stroškov. Indirektne koristi vključujejo nove izdelke in dejavnosti, prihranek časa za uporabnike aplikacij, ki uporabljajo odprte podatke, večjo učinkovitost v javnih storitvah in rast povezanih trgov. Predvideva se, da se bo med leti 2016 in 2020 ustvarilo 25.000 delovnih mest, direktno povezanih h odprtim podatkom [7]. Poleg koristi, povezanih s trgovino, ima uporaba odprtih podatkov še preostale prednosti. S statistično obdelavo odprtih podatkov na področju prometa in tranzita lahko identificiramo glavne dejavnike nesreč in tako zmanjšamo število umrlih na cestah. Ponovna uporaba podatkov lahko letno zmanjša do 5.5 % smrtnih žrtev na cestah letno [7].

1.2.1 Odprti podatki Slovenije

Politika odprtih podatkov v Sloveniji je v primerjavi z Evropsko unijo še v zaostanku. Glavni temelj politike odprtih podatkov v Republiki Sloveniji je Zakon o dostopu do informacij javnega značaja, ki omogoča, da lahko vsak državljan Republike Slovenije pridobi informacije javnega značaja. Javnost lahko dostopa do podatkov z vložitvijo zahteve, vendar čedalje več državnih organov podatke objavlja na svojih spletnih straneh. Eden največjih portalov za dostop do odprtih podatkov javnega sektorja je portal OPSI (Odprti

podatki Slovenije) [16].

Nacionalni portal odprtih podatkov OPSI je vzpostavilo Ministrstvo za javno upravo. Portal predstavlja celostni popis podatkovnih zbirk, ki jih vodijo organi javnega sektorja, in omogoča objavo zbirk v obliki odprtih podatkov. Portal OPSI je povezan s portalom Evropskih odprtih podatkov. OPSI ima dvojno funkcijo:

- Predstavlja centralni katalog evidenc in zbirk podatkov v državi, torej centralni popis metapodatkov o vseh evidencah in podatkovnih zbirkah, ki jih vodijo državni organi, občine in drugi organi javnega sektorja.
- Predstavlja enotno spletno mesto tudi za objavo podatkov iz zbirk v odprtih in strojno berljivih formatih. Kolikor je določena zbirka v odprtih formatih že objavljena na drugem spletnem mestu, je na portalu OPSI navedena spletna povezava na takšno spletno mesto [16].

Poleg dostopa do odprtih podatkov portal omogoča tudi dostop in objavo aplikacij, ki so nastale z uporabo odprtih podatkov.

Poleg tega je vredno omeniti tudi projekt opendata.si [17], v sklopu katerega je nastalo mnogo drugih projektov.

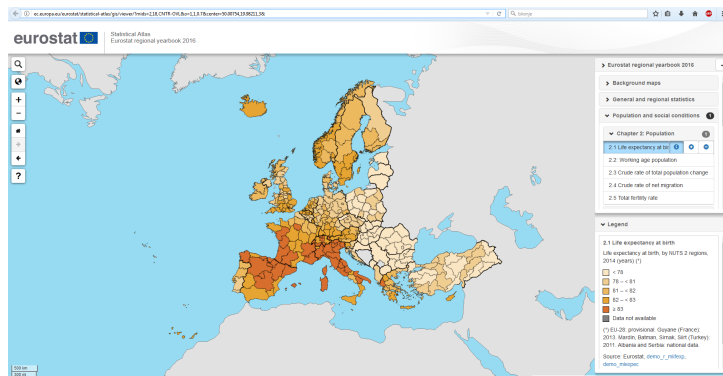
1.3 Vizualizacije odprtih podatkov

Z razvojem politike odprtih podatkov se je število vizualizacij skokovito povečalo. Z uporabo odprtih podatkov so nastale številne aplikacije. Najdemo jih lahko na straneh portala Evropskih odprtih podatkov.

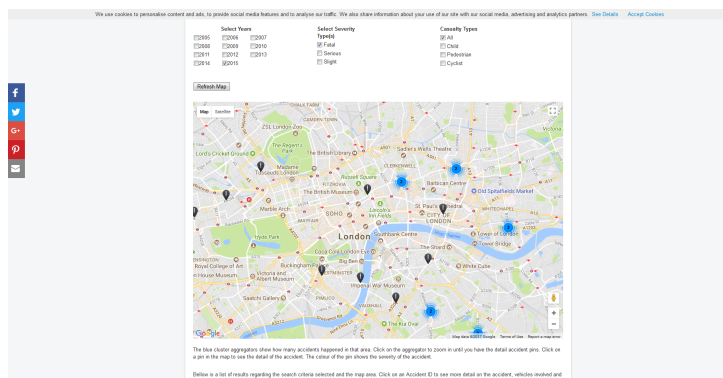
Ena zanimivejših aplikacij je Evropski statistični atlas [6] (glej sliko 1.1), ki prikazuje statistične in topografske zemljevide.

Primer vizualiziranja statističnih podatkov nesreč je spletni portal Road crash [22] (glej sliko 1.2).

Poleg aplikacij, ki vizualizirajo grobe statistične podatke, nastanejo tudi aplikacije, ki nam pomagajo pri odločitvah našega življenja. Aplikacija Is the



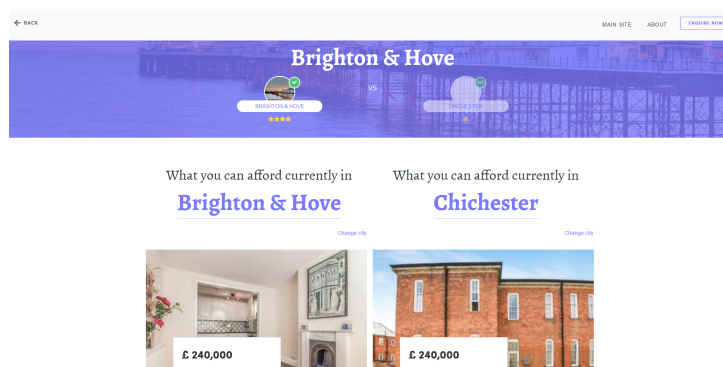
Slika 1.1: Eurostat statistični atlas.



Slika 1.2: Portal Road crash.

grass greener [11] (glej sliko 1.3) je aplikacija, ki omogoča primerjanje lokacij za bivanje v Veliki Britaniji.

Odprte podatke lahko uporabimo tudi v boju proti korupciji državnih organov. Projekt Red flag [20] je bil ustvarjen z namenom boja proti korupciji na Madžarskem.



Slika 1.3: Is the grass greener.

1.4 Vizualizacije odprtih podatkov na slovenskem področju

Kljub temu da se je odpiranje podatkov v Republiki Sloveniji začelo pozneje, je z uporabo odprtih podatkov nastalo že kar nekaj aplikacij.

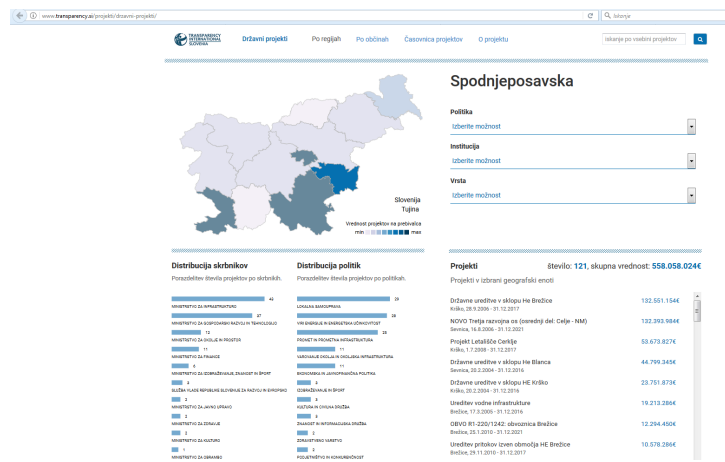
Portal plač javnega sektorja [19] je spletna aplikacija, ki omogoča pregled in analizo podatkov o tipih izplačil, vrstah izplačil in virih sredstev, ki jih proračunski uporabniki obračunavajo pri plačah zaposlenih v javnem sektorju [18].

Vizualizacija financiranja projektov iz državnega proračuna [29] (glej sliko 1.4) prikazuje geografsko porazdeljenost projektov glede na regije in občine Republike Slovenije. Omogoča tudi pregled projektov glede na časovno obdobje.

Vizualizacija proračun občin 2015 [33] (glej sliko 1.5) uporablja element sunburst [23] za hierarhičen prikaz proračuna občin za leto 2015 in omogoča primerjavo med dvema občinama.

Spletni portal Statist [37] (glej sliko 1.6) prikazuje podatke o oddanih javnih naročilih v Republiki Sloveniji.

Vizualizacija Proračun Republike Slovenije skozi čas [34] (glej sliko 1.7) prikazuje sestavo prihodkov oziroma odhodkov proračuna Republike Slovenije



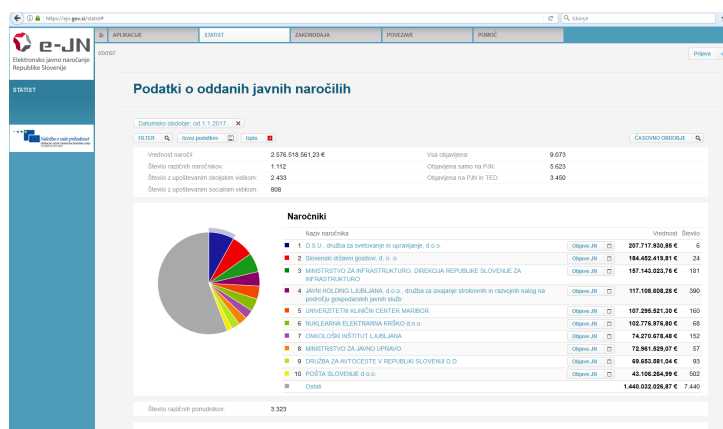
Slika 1.4: Vizualizacija projektov financiranje iz državnega proračuna.



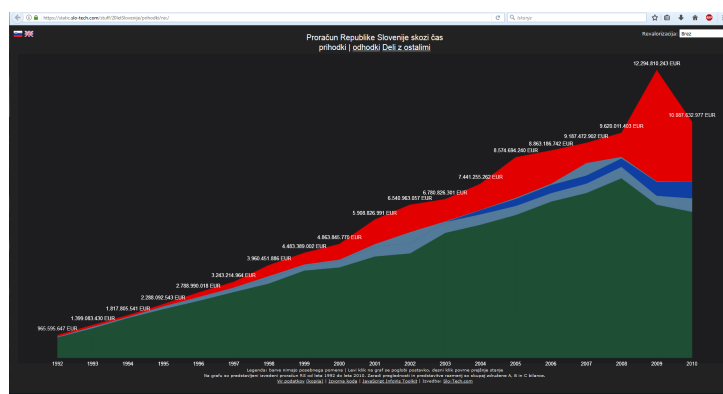
Slika 1.5: Vizualizacija proračun občin 2015.

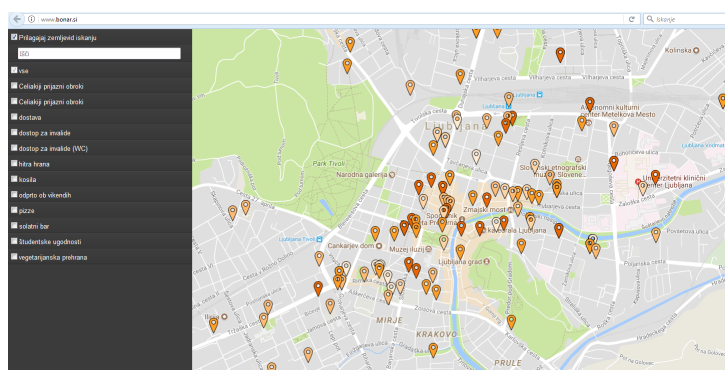
skozi čas.

Spletni portal Bonar [28] (glej sliko 1.8) je nastal za prikaz ponudnikov študentske prehrane.



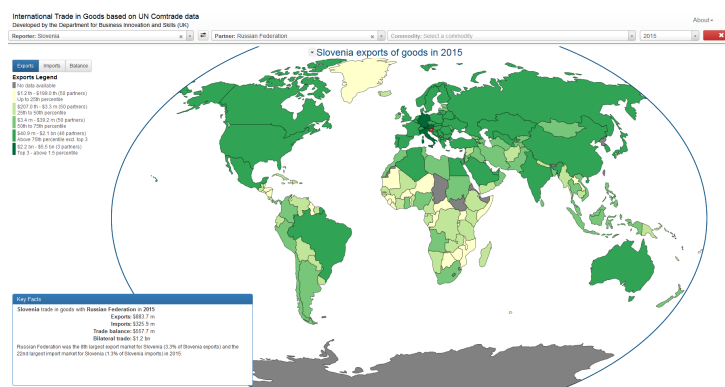
Slika 1.6: Spletni portal Statist.





Slika 1.8: Spletni portal Bonar.

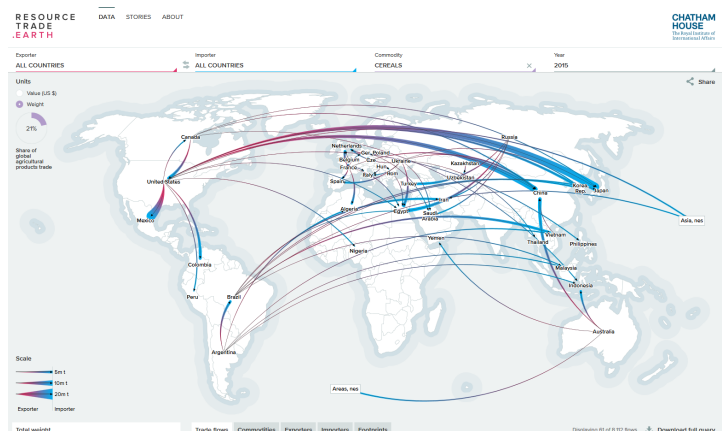
Vizualizacij podatkov na temo uvoz in izvoz je veliko, a kljub velikemu številu obstajajo faktorji, ki so skupni vsem vizualizacijam. Skupna tema vizualizacijam je prikaz pogleda izvoženega in uvoženega blaga glede na ostale države. Ena izmed pogostih rešitev je svetovni zemljevid držav. Države so med seboj lahko kategorizirane z barvami (glej sliko 1.9 [32]) ali pa drugimi elementi npr. puščicami (glej sliko 1.10 [35]).



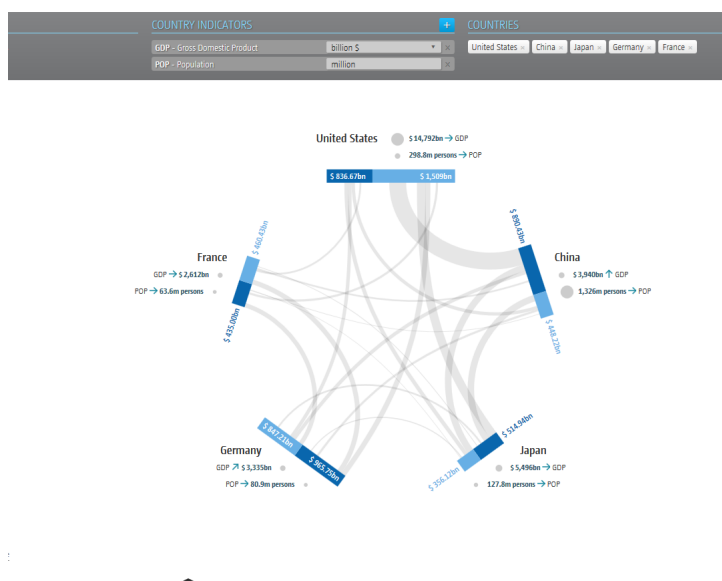
Slika 1.9: Vizualizacija International Trade in Goods based on UN Comtrade data.

Poleg zemljevida se pojavljajo tudi druge rešitve z uporabo grafov (glej sliko 1.11 [30]).

V pogledu posamezne države se uporabljajo grafikoni. Palični in stolpčni



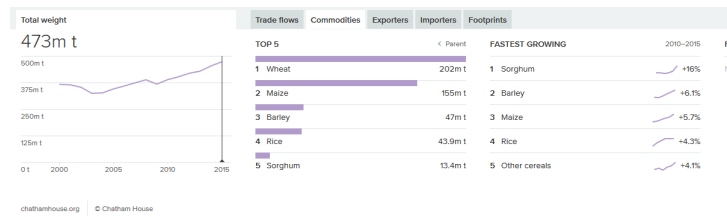
Slika 1.10: Vizualizacije Resource trade Earth.



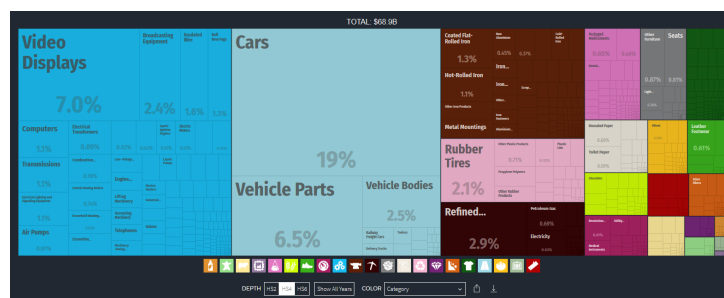
Slika 1.11: Primer vizualizacije z uporabo povezanega grafa.

grafikon se uporabljata za prikaz vrste blaga, črtni grafikon pa za prikaz gibanja skozi časovno obdobje (glej sliko 1.12 [35]). Za prikaz deleža in vrste blaga se uporablja tudi grafikon treemap (glej sliko 1.13 [36]).

Veliko vizualizacij ni omejenih na en pogled. Uporabljajo več pogledov oziroma prepuščajo izbiro uporabniku (glej sliko 1.14 [31]). Novejše vizualiza-

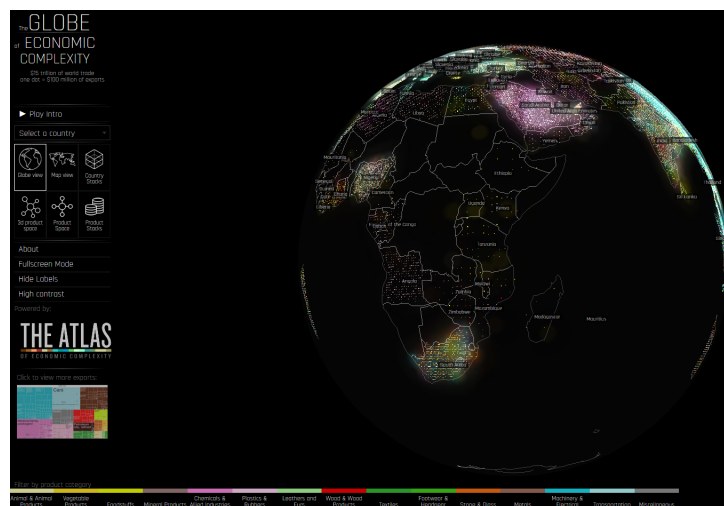


Slika 1.12: Vizualizacije Resource trade Earth.



Slika 1.13: Primer uporabe grafikona treemap.

cije imajo veliko stopnjo interaktivnosti in omogočajo uporabniku spreminjanje različnih parametrov in tako lažje ter zanimivejše raziskovanje podatkov.



Slika 1.14: Vizualizacija The globe of economic complexity.

1.6 Cilj

Ob pregledu vizualizacij na temo uvoza in izvoza, ki smo jih predstavili v prejšnjem poglavju, smo prišli do spoznanja, da je njihova največja pomanjkljivost ohlapnost kategorij blaga. Kot primer vizualizacije prikazujejo kategorijo blaga, npr. *Žive živali*, ne prikazujejo pa podatkov, ali gre za govedo, perutnino oziroma druge podkategorije. Ugotovili smo tudi, da v slovenskem prostoru konkretnije vizualizacije na to temo še ni. Zaradi teh razlogov smo se odločili razviti interaktivno vizualizacijo, katere cilj je prikazati količino in vrsto blaga, ki ga Republika Slovenija uvaža oziroma izvaža, ter prikazati razmerje med uvozom/izvozom po posameznih državah. Vizualizacija je bila zgrajena na podlagi konceptnega izdelka, ki je nastal kot končni projekt predmeta Multimedijske tehnologije.

Poglavje 2

Pregled tehnologij

V sodelovanju s Statističnim uradom Republike Slovenije smo definirali naslednje zahteve aplikacije:

- vizualizacija naj bo narejena v obliki spletne aplikacije,
- vizualizacija naj bo interaktivna,
- vizualizacija omogoča primerjavo držav,
- vizualizacija predstavi natančne kategorije podatkov.

Glede na zahtevo, da ustvarimo spletno aplikacijo, je bila uporaba tehnologij HTML5, CSS in JavaScript obvezna, saj predstavljajo daleč najbolj uporabljene standarde svetovnega spleta. Aplikacija je potrebovala podatkovno bazo, iz katere bi črpala podatke za vizualiziranje. Namesto klasičnih relacijskih podatkovnih baz smo se odločili za podatkovno bazo Elasticsearch. Razlogi za to odločitev so bili v obliki podatkov, naši podatki so predvsem tekstovni nizi, enostavnosti v dodajanju novih podatkov, hitrosti iskanja med podatki in dinamičnimi poizvedbami, ki jih podatkovna baza omogoča. Da bi poenotili izgled aplikacije, smo pri gradnji aplikacije uporabili ogrodje Bootstrap. Za izris grafičnega dela vizualizacije smo uporabili JavaScript knjižico D3.js. Knjižica D3.js je ena najbolj popularnih knjižic za izdelavo interaktivnih vizualizacij in grafikonov. Za uporabo le-te smo se odločili zaradi njene

odprtokodnosti in hitrosti, kar predstavlja pomembno lastnost, kadar imamo opravka z interaktivnostjo. V nadaljevanju bomo predstavili uporabljene tehnologije.

2.1 Iskalnik Elasticsearch

Iskalnik Elasticsearch je odprtokodni projekt, ki ga je začel Shay Banon. Je iskalnik, ki se ga zaradi velike hitrosti poslužujejo podjetja in korporacije, med drugimi tudi strani, kot so Wikipedia, Stack Overflow, GitHub in ostale [14, 10]. Zgrajen je bil na osnovi knjižice Apache Lucene, ki je ena najnovejših in najhitrejših iskalnih knjižic, napisana v programskem jeziku Java. Elasticsearch ni le iskalnik, lahko se ga opiše tudi kot:

- porazdeljena realnočasovna shramba dokumentov, kjer je vsako polje indeksirano in omogoča iskanje po njem,
- porazdeljeni iskalnik, ki omogoča analizo v realnem času,
- sposoben uporabljati na stotine strežnikov in petabajte strukturiranih in nestrukturiranih podatkov [10].

Končna aplikacija deluje kot samostojni strežnik, ki komunicira preko preprostega RESTful API-ja. Ker je objavljen pod licenco Apache 2, ga lahko prenašamo, uporabljamo in spreminjamo po svojih željah.

Shranjevanje v Elasticsearchu predstavlja korak stran od klasičnih relacijskih baz. Entitete, kot so tabela, vrstica, relacija nadomestijo:

Indeks:

Prostor, kjer se shranjujejo podatki in je ekvivalent tabelam v relacijskih bazah [14] .

Dokument:

Je glavna entiteta shranjevanja pri Elasticsearchu. V relacijskih bazah je ekvivalent vrstici v tabeli. Dokumenti so sestavljeni iz polj. Vsako polje ima tip (tekst, število, datum ...), lahko pa vsebuje tudi bolj kompleksne

vrednosti, kot so tabele ali pa poddokumenti. Tip polja ima pomembno funkcijo, saj definira, kako se nad njim izvedejo operacije, kot sta primerjava in sortiranje. Za razliko od relacijskih baz dokumentom ni potrebno imeti fiksne strukture, vsak dokument ima lahko različna polja [14].

Tip dokumenta:

Indeks lahko vsebuje veliko objektov za različno uporabo. Na primer aplikacija za spletni blog bo shranjevala komentarje, vizualizacija podatkov pa statistične podatke. Tip dokumenta omogoča lažje razlikovanje med temi objekti [14].

Vozlišče in gruča (node and cluster):

Elasticsearch lahko deluje kot samostojni strežnik ali pa kot skupina strežnikov, ki medsebojno sodelujejo. Tej skupini pravimo gruča, vsak strežnik pa predstavlja eno vozlišče. S povezovanjem strežnikov v gruča lahko porazdelimo količino podatkov med posamezna vozlišča in tako pridobimo na hitrosti in odzivnosti [14].

Drobec (shard):

Včasih pridemo do situacije, kjer zaradi velikega števila dokumentov, eno vozlišče ni dovolj, bodisi zaradi premajhne količine delovnega pomnilnika, bodisi zaradi premajhne procesorske moči. V takšnem primeru dokumente razbijemo na več kosov (drobcev), kjer vsak drobec postane nov indeks. Tako lahko drobce porazdelimo med strežnike. Takšna porazdelitev podatkov je popolnoma transparentna za aplikacijo, ki uporablja Elasticsearch [14].

Kopija:

Kopije drobcev uporabimo za povečanje razpoložljivosti podatkov ali pa za povečanje zmognosti poizvedovanja [14].

Za komuniciranje s strežnikom Elasticsearch obstaja več vmesnikov za namensko programiranje. Najpogostejše uporabljena vmesnika sta Java in RESTful, ki je v nadaljevanju tudi bolj podrobno opisan.

Vmesnik RESTful sloni na spletni storitvi REST, ki s strežnikom Elasticsearcha komunicira preko porta 9200. Zahteva na strežnik je sestavljena tako kot katera druga zahteva protokola HTTP.

curl -X ZAHTEVA ' PROTOKOL :// GOSTITELJ / POT ? POIZVEDBENI NIZ' -d ' POIZVEDBA '

Zahteva:

Zahteva lahko vsebuje metode GET, POST, PUT, HEAD ali DELETE.

Protokol:

Lahko uporabimo protokol HTTP ali HTTPS.

Gostitelj:

Naslov vozlišča v gruči ali lokalni naslov vozlišča na lokalnem računalniku.

Vrata:

Vrata, ki jih storitev Elasticsearch uporablja. Privzeta so 9200.

Poizvedbeni niz:

Vsebuje dodatne poizvedbene kriterije (na primer *?pretty* povzroči, da je odgovor v lažji bralni obliki).

Telo:

Vsebuje poizvedbo v formatu JSON.

Glede na uspešnost izvedene poizvedbe in vrsto zahteve Elasticsearch vrne odziv HTTP [10].

2.2 Format JSON

JSON (JavaScript Object Notation) je tekstovni format za serializacijo strukturiranih podatkov. Izvira iz zapisa objektov v programskem jeziku JavaScript. Uporablja se tudi kot standard za izmenjavo podatkov med aplikacijami.

JSON lahko predstavi štiri primitivne tipe podatkov (znakovna polja, števila, logično vrednost in vrednost *null*) in dva strukturirana tipa (objekt in tabela) [4].

JSON ima naslednjo strukturo:

Objekt: Osnovni element je objekt, ki se začne z znakom `{`, konča z `}` in vsebuje par, ime in vrednost, ki sta ločena z znakom `:`. Primer JSON objekta: `{"avto":"Fiat"}`.

Vrednost: Polje vrednost lahko vsebuje katerikoli podatkovni tip.

Ime: Vsebuje znakovno polje (*String*).

Tabela: Je predstavljena z znakoma [in] in vsebuje elemente, ločene z ,.
Format JSON privzeto uporablja kodiranje *UTF-8*.

2.3 Knjižica D3.js

D3.js (Data-Driven Documents) je JavaScript knjižica za izdelavo dinamičnih in interaktivnih vizualizacij in je trenutno vodilno orodje za izdelavo spletnih vizualizacij. Ustvaril jo je Mike Bostock kot naslednika orodij za vizualizacijo Protovis. Omogoča nam povezovanje poljubnih podatkov na DOM (Document Object Model) in uporabo podatkovno gnanih transformacij nad dokumenti. Dokumente manipulira na podlagi podatkov z uporabo standardov HTML, CSS in SVG [38, 5].

2.4 Bootstrap

Pri izdelavi aplikacije smo uporabili ogrodje Bootstrap. Bootstrap je najbolj popularno odprtokodno ogrodje za izdelavo spletnih vmesnikov aplikacij. Vsebuje predloge, narejene na podlagi HTML in CSS, kot so gumbi, tipografija, obrazci in druge komponente vmesnikov. Bootstrap podpira večino modernih brskalnikov, kot so Firefox, Chrome, Internet Explorer10+, Microsoft Edge in Opera. Zaradi uporabe odzivnega spletnega dizajna Bootstrap podpira tudi brskalnike mobilnih naprav. Je modularen in je sestavljen iz serij *Less* datotek s slogi. S prihodom Bootstrap 4 bo standard za datoteke s slogi LESS zamenjal standard *Sass*.

Poglavje 3

Vizualizacija mednarodne blagovne menjave v Sloveniji

Kot sem že omenil v uvodu, je bila aplikacija razvita na osnovi končnega izdelka pri predmetu Multimedijske tehnologije. Ob skrbnem mentorstvu in v sodelovanju s Statističnim uradom Republike Slovenije smo nadgradili in razvili spletno vizualizacijo uvoza in izvoza Slovenije. Tako lahko na interaktiven in uporabniku prijazen način prikažemo širši javnosti količino in vrsto blaga, ki ga Republika Slovenija uvaža oziroma izvaža. Pri izdelavi aplikacije so bile uporabljene tehnologije JavaScript, HTML5, CSS3, Java in Elasticsearch.

3.1 Podatki

Podatke, ki jih aplikacija uporablja, je priskrbel Statistični urad Republike Slovenije. Podatki so javno dostopni in so dosegljivi preko spleta [24].

3.1.1 Vrsta podatkov

Uporabljeni so bili podatki o državah oziroma območjih ter letu, vrsti in količini blaga uvoza in izvoza.

Meritve: Uporabljeni sta bili vrednosti *izvoz* in *uvoz*.

Mesec: Uporabljeni so bili podatki od leta 2000 do aprila 2017.

Država: Uporabljenih je bilo vseh 255 območij.

Enota: Uporabljeni sta bili vrednosti *1000 EUR* in *Neto masa v 1000 kg*.

Kombinirana nomenklatura: Uporabljene so bile vse vrednosti do nivoja *tarifna skupina HS, KN*.

3.1.2 Obdelava in shranjevanje podatkov

Podatki so bili preneseni v obliki datotek *csv* [21], kjer so vrednosti ločene z znakom *;*. Ker je v aplikacijo smiselno podatke vnašati glede na leto, je vsako leto predstavljalo eno datoteko *csv*.

Primer vrstice v datoteki:

```
"Nemčija";"1000 EUR";"Kombinirana nomenklatura { SKUPAJ";4389433;4181848
```

Podatke je bilo potrebno pretvoriti v format JSON, da so bili primerni za uvoz v iskalnik Elasticsearch. Za obdelavo je bila ustvarjena skripta v programskem jeziku Java.

Skripta kot argument sprejme datoteko *csv* s podatki, jo prebere in obdelava tako, da je končni rezultat primeren za uvoz v Elasticsearch z vmesnikom BULK. Ker je bilo smiselno, da se ohrani hierarhija kombinirane nomenklature, je bilo potrebno nadomestiti imena kategorij z njihovimi kodami. Kot primer kategorija *Žive živali* ima kodo *02*, njene podkategorije pa *0201*, *0202*... Primer končnega rezultata:

```
{ "index" : { "_index" : "drzave", "_type" : "drzave" } }
{"drzava":"Združeni arabski emirati","uvoz":true,"leto":2015,"teza":false,
"00":4353,"01":0,"0101":0,...}
```

Za podatkovno bazo smo se odločili za tekstovni iskalnik Elasticsearch, saj je bil primeren zaradi hitrosti, poleg tega pa omogoča precej dinamične poizvedbe. Končen rezultat skripte smo uvozili v Elasticsearch različica 5.3.2 v indeks *drzave*.

Da pa bi še vedno imeli podatke o imenih kategorij, smo ustvarili dodaten indeks *blago*. Vanj smo uvozili podatke o imenih kategorij kombinirane nomenklature in njihovih kod. Primer podatka, uvoženega v indeks *blago*.

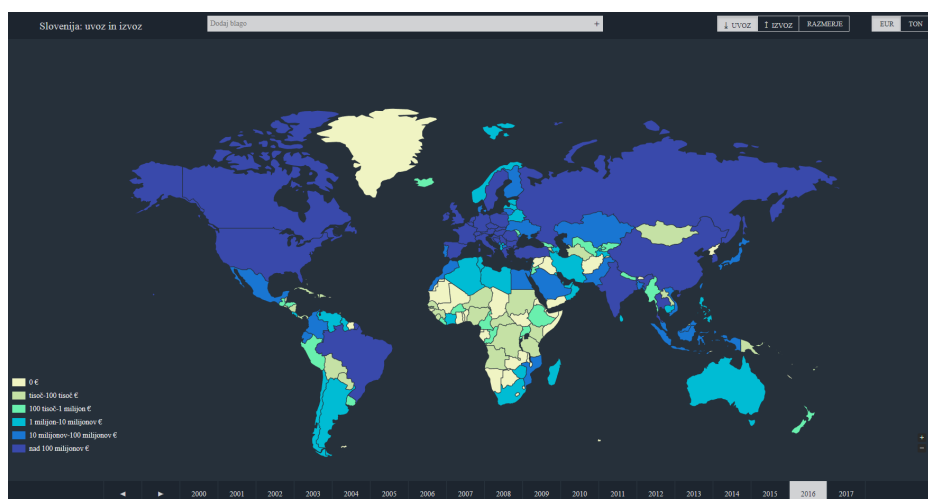
```
{ "index" : { "_index" : "blago", "_type" : "blago" } }  
{"blago":"Žive živali","koda":"01"}
```

3.2 Vizualizacija uvoz in izvoz Slovenije

Kot že omenjeno, je aplikacija nadaljevanje projekta pri predmetu Multi-medijske tehnologije. Gre za spletno vizualizacijo na temo uvoz in izvoz Republike Slovenije. Izdelana je s tehnologijami JavaScript, HTML5, CSS3, jQuery, ogrodje Bootstrap 3 in knjižico D3.

3.2.1 Pogled zemljevid

Začetni pogled aplikacije prikazuje (slika 3.1) interaktivni zemljevid, ki glede na barvo in položaj miške prikazuje informacije.



Slika 3.1: Pogled zemljevid.

V zgornjem delu pogleda imamo meni gumbov in tekstovnih polj, s katerimi spreminjamo parametre aplikacije. Tukaj lahko uporabnik spreminja zelene parametre glede na vrsto blaga, smer toka blaga in enote blaga. Vsi elementi menija so oblikovani oziroma izdelani s pomočjo ogrodja Bootstrap 3 [2].

Tekstovno polje: Gre za tekstovno polje s funkcijo samodokončanja [27], kjer vanj vpisujemo vrsto blaga, ki ga iščemo. Privzeta vrsta je *Kombinirana nomenklatura-SKUPAJ*, kar pomeni vsoto vseh vrst blaga. Ob kliku na gumb $+$, ki se nahaja na skrajni desni strani polja, se nam prikaže seznam vseh vrst blaga (slika 3.2).

Skupina gumbov UVOZ, IZVOZ, RAZMERJE: Z njimi spreminjamo, ali želimo prikazati podatke o uvozu, izvozu ali razmerju med uvozom in izvozom.

Skupina gumbov EUR, TON: Z gumboma spreminjamo, ali želimo podatke v enoti *1000 EUR* ali *Neto masa v 1000 kg*.

Glavni element pogleda je zemljevid choropleth sveta. Za ta prikaz smo se odločili, saj imamo opravka s podatki, ki so vezani na regije, v našem primeru države sveta. Uporabili smo predlogo zemljevida, izdelanega z JavaScript knjižnico JVectorMap [13], in ga poslovenili. Zemljevid prikazuje države sveta, ki se glede na parametre obarvajo po dobljenih vrednostih. Za pomoč in lažje razlikovanje je v levem spodnjem kotu tudi legenda. Barve zemljevida in legende so bile izbrane iz barvne palete *material design*. Da bi uporabnik imel dostop tudi do numeričnih vrednosti, se ob pozicioniranju miškega kazalca na regijo pojavi okno, ki vsebuje matematični zapis vrednosti regije.

V skrajnem spodnjem delu pogleda se nahaja časovnica, s katero nadzorujemo podatke glede na leto. Časovnico sestavlja element *Bootstrap Navigation Bar*, ki smo ga vizualno spremenili za naše potrebe. Časovnica se ustvari ob zagonu aplikacije glede na podatke, ki jih dobi iz podatkovne baze Elasticsearch. Razlog za takšen pristop je, da v primeru dodajanja podatkov uvoza in izvoza za prihodnja leta v bazo ni potrebno spreminjati programske kode aplikacije.

3.2.2 Sezam filtrov in filtri

Ko imamo opravka z večjim številom kategorij podatkov, je smiselno, da ima uporabnik možnost filtriranja želene kategorije. V našem primeru, če uporabnika zanima samo blago *Farmaceutski proizvodi*, naj ima možnost, da si to tudi ogleda. V aplikaciji to izražamo s filtri. Filtri predstavljajo dodatno funkcijo aplikacije, s katerimi lahko dodatno spreminjamo parametre o vrsti blaga. Filter dodamo ali z vpisom zelenega blaga v tekstovno polje ali s klikom na želeno blago v seznamu blaga. Ob tem se v aplikaciji prikaže meni filtrov (slika 3.3). Podvajanje filtrov ni mogoče. Filter lahko odstranimo s klikom na oznako *X*, ki se nahaja levo od imena filtra. Aplikacija omogoča dodajanje treh filtrov naenkrat, saj se za vsak dodan filter količine blaga seštevajo. Kot primer, če dodamo filter *Žive živali*, aplikacija prikaže količine za blago *Žive živali*. Če dodamo še filter *"Farmaceutski proizvodi"*, aplikacija prikaže vsoto količine obeh blagov.

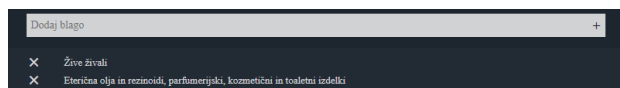
Uporabnik mora imeti tudi možnost priti do informacije, kakšne vrste blaga sploh obstajajo. V ta namen ob kliku na gumb *+*, ki se nahaja na skrajni desni strani polja, se nam prikaže drevesni seznam [25] vseh vrst blaga (slika 3.2). Začetni pogled prikazuje *poglavja HS, KN kombinirane nomenklature*, ob kliku na *+* pa se želeno poglavje razširi na njegove *tarifne skupine HS, KN kombinirane nomenklature*. Ob kliku na ime blaga blago dodamo v filter (slika 3.3). Pogled seznama filtrov zapremo s klikom na ime blaga ali na gumb *X*.

3.2.3 Pogled države.

V pogledu zemljevida smo prikazali primerjavo uvoza in izvoza blaga Republike Slovenije med državami sveta. Kaj pa, če nas zanima samo določena država. Rešitev tega vprašanja predstavlja pogled države. Ob kliku na državo v pogledu *zemljevid* se nam prikaže pogled države (slika 3.4). Časovnica in meni za izbiro parametrov ostaneta ista, odstrani se le skupina gumbov *UVOZ, IZVOZ, RAZMERJE*, saj v tem pogledu ni smiselna.



Slika 3.2: Seznam filtrov.



Slika 3.3: Primer filtrov.

Spreminjanje spremenljivke skozi čas je ena najbolj zanimivih lastnosti pri vizualizaciji podatkov. V povezavi s temo aplikacije je bilo zelo smiselno to lastnost tudi predstaviti. Eden najboljših načinov za predstavitev gibanja skozi čas je črtni graf, zato je uporabljen tudi tukaj. V aplikaciji črtni graf predstavi gibanje uvoza in izvoza skozi čas za izbrano blago in državo. Graf je izdelan s pomočjo orodja Google Charts [9].

Za predstavitev razmerja med količinami imamo veliko možnosti. Uporabimo lahko stolpične, tortne, palične grafikone in še kaj. Ko hočemo predstaviti več vrednosti v medsebojnem razmerju, sta najboljša rešitev stolpični ali palični grafikon. V aplikaciji smo razmerje med uvozom in izvozom predstavili z dvema nasproti ležečima paličnima grafikonom. Zaradi daljših imen kategorij blaga je bila uporaba paličnega grafikona bolj smiselna, saj imamo več prostora za oznake kot s stolpičnim grafom. Levi palični grafikon prika-

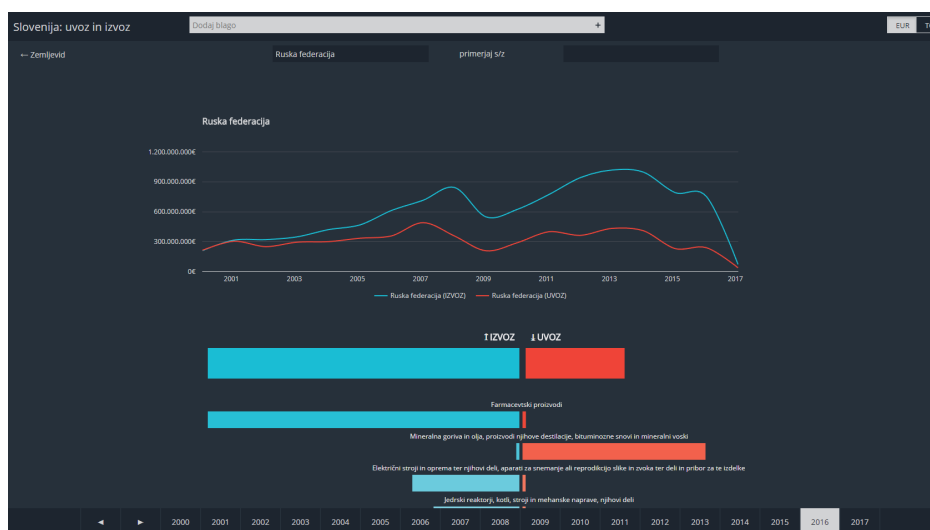
zuje izvoz, desni pa uvoz. Ob premiku miškega kazalca na zelen grafikon se pojavi okno z zapisom količine, ki jo grafikon predstavlja.

Ko prikazujemo nek del podatkov, v našem primeru del uvoza ali izvoza, je zaželeno, da podamo tudi informacijo, kolikšen del celote predstavlja. Potrebno je bilo vizualizirati razmerje med skupnim uvozom/izvozom, razmerje med skupnim uvozom/izvozom in določeno vrsto blaga ter razmerje med posameznimi vrstami blaga. Palične grafikone smo razdelili v dve skupini. Zgornji večji par paličnih grafikonov prikazuje skupen uvoz oziroma izvoz izbrane države. Če nimamo izbranih filtrov, spodnji manjši pari paličnih grafikonov prikazujejo po količini največje menjave blaga, in sicer deset vrst blaga za izbrano državo. V primeru, da imamo izbrane filtre, večji par paličnih grafikonov prikazuje seštete količine *poglavj HS, KN kombinirane nomenklature* na izbrane filtre, manjši pari pa *tarifne skupine HS, KN kombinirane nomenklature* izbranih filtrov. Ob pozicioniranju miškega kazalca na manjši par paličnih grafikonov določenega blaga se na večjem paličnem grafikonu izriše procentualna vrednost med celoto in določenim blagom.

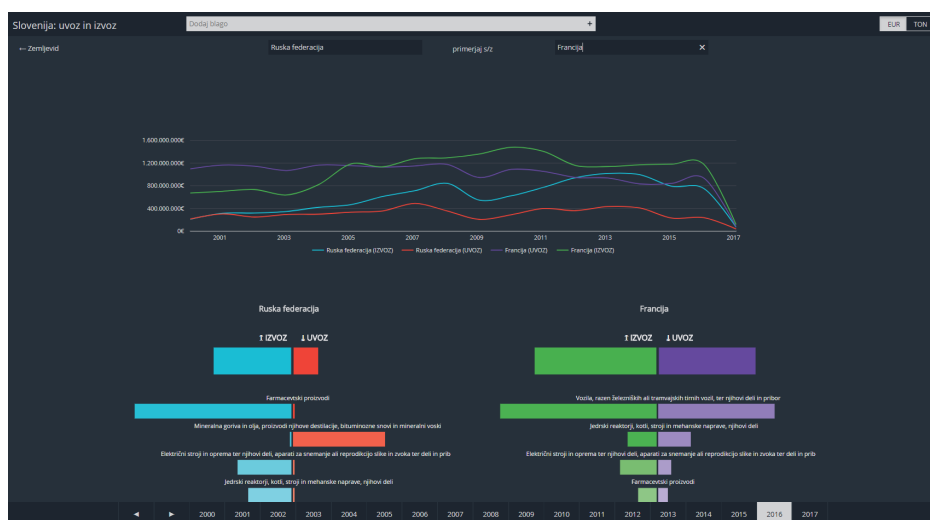
Dolžina para paličnih grafikonov, ki predstavljata skupen uvoz in izvoz, je normalizirana glede na največjo vrednost obeh parov. Dolžina spodnjih grafikonov pa je normalizirana na največjo vrednost količine med njimi.

Kot smo že spoznali, dobra vizualizacija mora omogočati čim več primerjav med podatki. Države primerjamo med seboj že v pogledu zemljevida, vendar ta pogled ne omogoča primerjave držav po npr. sestavi uvoza. Zato ima pogled države možnost primerjave izbrane države z drugo državo. Želena državo vpišemo v tekstovno polje ali jo izberemo s seznama predlogov in izrišejo se dodatni grafikoni (glej sliko 3.5). Dodano državo lahko odstranimo s klikom na oznako *X*, ki se nahaja desno od imena dodane države.

Za prehod med pogledoma smo uporabili gumb \leftarrow *Zemljevid*. S klikom nanj se vrnemo na pogled zemljevida.



Slika 3.4: Pogled države.



Slika 3.5: Primer primerjave držav.

3.2.4 Skripta za dodajanje podatkov

Ena izmed dobrih praks razvijanja aplikacij je pripravljenost aplikacije na prihodnost – nadgradljivost. Izdelana vizualizacija uporablja podatkovno bazo, v kateri so že podatki do letošnjega leta. Da bi vizualizacija bila

aktualna tudi čez nekaj let, je zaželeno, da se v podatkovno bazo uvažajo novi podatki. V ta namen je bila izdelana skripta, ki to nalogo precej olajša.

Programska skripta je napisana v programskem jeziku Java. Uporablja upravljalca projektov programske opreme *Maven* [15]. Za dostop do strežnika Elasticsearch je uporabljen *Elasticsearch Java API*.

Skripta sprejme tri argumente, lokacijo datotke .csv, ki vsebuje podatke, leto, za katero uvažamo podatke, in naslov strežnika Elasticsearch. Skripta na začetku preveri, če datoteka obstaja oziroma je pravilne oblike in če smo podali smiselno leto. Nato ustvari povezavo do strežnika Elasticsearch z uporabo razreda *TransportClient*. Ker obstaja možnost, da podatkov za določeno leto ne uvažamo prvič, ampak jih hočemo samo posodobiti, skripta pobriše podatke za tisto leto iz baze Elasticsearch. To naredi z uporabo razreda *DeleteByQueryAction*. Del kode, kjer se povežemo s strežnikom Elasticsearch in pobrišemo podatke za določeno leto:

```
TransportClient client = new PreBuiltTransportClient(Settings.EMPTY);

client.addTransportAddress(new InetSocketAddressTransportAddress (
InetAddress.getByAddress(streznikElasticsearch), 9300))
.addTransportAddress(new InetSocketAddressTransportAddress(
InetAddress.getByAddress(streznikElasticsearch), 9300));

BulkIndexByScrollResponse response =
DeleteByQueryAction.INSTANCE.newRequestBuilder(client)
.filter(QueryBuilders.matchQuery("leto", leto))
.source("drzave")
.get();
```

Ko je povezava vzpostavljena in so podatki pobrisani, skripta prebere datoteko in obdela podatke tako, da jih pretvori v smiselno obliko JSON. Za konec sledi še shranjevanje podatkov:

```
IndexResponse response1 = client.prepareIndex("drzave", "drzave")
```

```
.setSource(uvoz)
.get();
response = client.prepareIndex("drzave", "drzave")
.setSource(izvoz)
.get();
```

3.2.5 Težave in rešitve

Pri realizaciji aplikacije smo naleteli na nekaj težav. Prvi problem, na katerega smo naleteli, je bil prikaz aplikacije na mobilnih napravah. Ker je aplikacija primarno namenjena namiznim računalnikom, se za interakcijo z elementi uporabljata dogodka *mouseover* in *mouseout*, ki povzročita interakcijo ob prehodu miškega kazalca čez element. Pri mobilnih napravah interakcija s temi dogodki ni mogoča. V pogledu države je bila rešitev implementirana z uporabo dogodkov *touchstart* in *touchend*, ki omogočita enako interakcijo, kot sta jo povzročila dogodka *mouseover* in *mouseout*. V pogledu zemljevid te rešitve ni bilo mogoče implementirati, saj knjižica JVector map dogodkov *touchstart* in *touchend* ne podpira. Sklenjen je bil kompromis, da za mobilne naprave te interaktivnosti v pogledu zemljevida ne bo.

Naleteli smo tudi na težavo pri združljivosti aplikacije s starejšimi brskalniki. Starejše verzije brskalnikov, npr. Mozilla Firefox 40.0.2, ne podpirajo uporabe funkcije `array.includes(string)`. Težavo smo rešili z uporabo funkcije `array.indexOf(string)`.

Med realizacijo aplikacije smo ugotovili, da aplikacija ni pravilno delovala, ko je do nje hotel dostopati Statistični urad Republike Slovenije. Razlog je bil v varnostni politiki Statističnega urada, saj ima njihovo omrežje odprta samo vrata 80, aplikacija pa je uporabljala tudi vrata 9200 za dostop do podatkovne baze Elasticsearch. V ta namen smo se odločili implementirati obratni posredniški strežnik (reverse proxy server), ki je omogočil, da uporabnik za dostop do aplikacije uporablja samo vrata 80, za komunikacijo aplikacije do baze pa poskrbi obratni posredniški strežnik.

Poglavje 4

Sklepne ugotovitve

Kot smo ugotovili, je področje vizualiziranja podatkov v Sloveniji, v primerjavi z Evropsko unijo, še v razvoju. Vendar z razvojem področja odprtih podatkov se bo pospešil tudi razvoj vizualiziranja. Tako bomo lahko hitreje raziskovali in predstavili podatke državnih organov. S tem bomo lahko odkrili napake in področja, potrebna izboljšav, ter jih odpravili.

Cilj naloge je bil prispevati k širjenju področja vizualizacije in uporabe odprtih podatkov. Ustvarili smo interaktivno vizualizacijo podatkov na temo uvoza in izvoza Republike Slovenije. Vizualizacija je uporabnikom prijazna, omogoča hiter in jasen pregled gibanja uvoza in izvoza ter vrst blaga Republike Slovenije skozi leta. Poleg tega pa je pripravljena na prihodnost, saj smo omogočili enostaven vnos novih podatkov.

Kljub uspešnosti izdelka obstajajo možnosti nadgradnje. Zanimiva bi bila povezava obstoječih uporabljenih podatkov z drugimi bazami podatkov, npr. kam izvažajo/uvažajo slovenska mikropodjetja. Možno bi bilo tudi dodati dodatne poglede, npr. odvisnost med vrstami blaga, če iz določene države uvažamo/izvažamo blago 1, kakšna je možnost, da uvažamo tudi blago 2. Možnosti je veliko, omejuje nas samo naša domišljija.

Literatura

- [1] Jacques Bertin. Semiology of graphics: diagrams, networks, maps. 1983.
- [2] Bootstrap. Dosegljivo: <http://getbootstrap.com/>. [Dostopano: 1. 8. 2017].
- [3] Chun-houh Chen, Wolfgang Karl Härdle, in Antony Unwin. *Handbook of data visualization*. Springer Science & Business Media, 2007.
- [4] Douglas Crockford. The application/json media type for javascript object notation (json). 2006.
- [5] D3.js. Dosegljivo: <https://d3js.org/>. [Dostopano: 1. 8. 2017].
- [6] European statistical atlas. Dosegljivo: <http://ec.europa.eu/eurostat/statistical-atlas/gis/viewer/>. [Dostopano: 1. 8. 2017].
- [7] European Union open data. Dosegljivo: https://www.europeandataportal.eu/sites/default/files/edp_creating_value_through_open_data_0.pdf. [Dostopano: 1. 8. 2017].
- [8] Finančni slovar. Dosegljivo: <http://www.financnislovar.com/definicije/bruto-domaci-proizvod.html>. [Dostopano: 1. 8. 2017].
- [9] Google Charts. Dosegljivo: <https://developers.google.com/chart/>. [Dostopano: 1. 8. 2017].
- [10] Clinton Gormley in Zachary Tong. *Elasticsearch: The Definitive Guide: A Distributed Real-Time Search and Analytics Engine*. O'Reilly Media, Inc., 2015.

-
- [11] Is the grass greener. Dosegljivo: <http://www.onlinemortgageadvisor.co.uk/is-the-grass-greener/search/>. [Dostopano: 1. 8. 2017].
- [12] Julie Iliinsky, Noah in Steele. *Designing data visualizations: representing informational relationships*. O'Reilly Media, Inc., 2011.
- [13] jvectormap. Dosegljivo: <http://jvectormap.com/maps/world/world/>. [Dostopano: 1. 8. 2017].
- [14] Marek Kuc, Rafal in Rogozinski. *ElasticSearch server*. Packt Publishing Ltd, 2013.
- [15] Maven. Dosegljivo: <https://maven.apache.org/>. [Dostopano: 1. 8. 2017].
- [16] Odprti podatki Slovenije. Dosegljivo: <https://podatki.gov.si/>. [Dostopano: 1. 8. 2017].
- [17] Opendata.si. Dosegljivo: <https://opendata.si/>. [Dostopano: 1. 8. 2017].
- [18] Plače javnega sektorja. Dosegljivo: <https://podatki.gov.si/aplikacije/portal-pla%C4%8D-javnega-sektorja>. [Dostopano: 1. 8. 2017].
- [19] Portal plač JS. Dosegljivo: <http://www.pportal.gov.si/>. [Dostopano: 1. 8. 2017].
- [20] Red Flags. Dosegljivo: <http://www.redflags.eu/>. [Dostopano: 1. 8. 2017].
- [21] RFC 4180. Dosegljivo: <https://tools.ietf.org/html/rfc4180>. [Dostopano: 1. 8. 2017].
- [22] Road crash maps. Dosegljivo: <https://www.roadcrash.co.uk/map>. [Dostopano: 1. 8. 2017].

-
- [23] Sunburst chart. Dosegljivo: <https://www.cc.gatech.edu/gvu/ii/sunburst/>. [Dostopano: 1. 8. 2017].
- [24] Izvoz in uvoz po 2-, 4- in 6-mestni šifri Kombinirane nomenklature in po državah, Slovenija, kumulativni podatki. Dosegljivo: http://pxweb.stat.si/sistat/MainTable/tbl_2401742. [Dostopano: 1. 8. 2017].
- [25] Treeview. Dosegljivo: <https://github.com/jonmiles/bootstrap-treeview>. [Dostopano: 1. 8. 2017].
- [26] Edward Tufte. *The visual display of quantitative information*. Graphics Press, 1983.
- [27] Typeahead. Dosegljivo: <https://github.com/bassjobsen/Bootstrap-3-Typeahead>. [Dostopano: 1. 8. 2017].
- [28] Spletni portal Bonar. Dosegljivo: <http://www.bonar.si/>. [Dostopano: 1. 8. 2017].
- [29] Financiranje državnih projektov. Dosegljivo: <http://www.transparency.si/projekti/drzavni-projekti/>. [Dostopano: 1. 8. 2017].
- [30] Project GED VIZ. Dosegljivo: <https://viz.ged-project.de/>. [Dostopano: 1. 8. 2017].
- [31] The globe of economic complexity. Dosegljivo: <http://globe.cid.harvard.edu/?mode=productsphere&id=CL>. [Dostopano: 1. 8. 2017].
- [32] International trade in goods based on UN Comtrade data. Dosegljivo: <https://comtrade.un.org/labs/BIS-trade-in-goods/?reporter=705&partner=643&year=2015&flow=2>. [Dostopano: 1. 8. 2017].
- [33] Proračun občin 2015. Dosegljivo: <http://www.transparency.si/projekti/proracuni-obcin/>. [Dostopano: 1. 8. 2017].

- [34] Proračun Republike Slovenije. Dosegljivo: <https://static.slo-tech.com/stuff/20letSlovenije/odhodki/nic/>. [Dostopano: 1. 8. 2017].
- [35] Resource trade Earth. Dosegljivo: <https://resourcetrade.earth/data?year=2015&category=7&units=weight>. [Dostopano: 1. 8. 2017].
- [36] Slovakia exports, imports and trade partners. Dosegljivo: <http://atlas.media.mit.edu/en/profile/country/svk/>. [Dostopano: 1. 8. 2017].
- [37] Spleti portal Statist. Dosegljivo: <https://ejn.gov.si/statist>. [Dostopano: 1. 8. 2017].
- [38] Nick Qi Zhu. *Data visualization with D3.js cookbook*. Packt Publishing Ltd, 2013.